U.S. Army Research Laboratory
SUMMER RESEARCH TECHNICAL REPORT

Motion Simulation in the Environment for Auditory Research

BRAXTON BOREN
MENTOR: MARK ERICSON
HUMAN RESEARCH AND ENGINEERING DIRECTORATE
ABERDEEN PROVING GROUND, MARYLAND

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **AUG 2011**	2. REPORT TYPE	3. DATES COVERED **00-00-2011 to 00-00-2011**

4. TITLE AND SUBTITLE **Motion Simulation In The Environment For Auditory Research**	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Human Research And Engineering Directorate,Aberdeen Proving Ground,MD,21005**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
See also ADA548876

14. ABSTRACT
Virtual sound source motion has been implemented in the U.S. Army Research Laboratory?s Environment for Auditory Research, which contains a 57-channel spherical loudspeaker array located in a semi-anechoic chamber. Using the low-latency PortAudio application programming interface from the Psychophysics Toolbox Version 3, we are able to dynamically update 57 channels of streaming audio in real time using MATLAB for signal processing. Both Distance-Based Amplitude Panning (DBAP) and Vector Base Amplitude Panning (VBAP) have been implemented in MATLAB for controlling source motion. Sources are defined on a given path, such as a circle, ellipse, or the ?dog bone? pattern often used in aviation. Although DBAP works convincingly for virtual sources located on the sphere defined by the loudspeaker array, VBAP is needed to position sources outside the array. Source motion paths are defined parametrically with respect to time, and the playback buffer updates the panned position every 11.5 ms. Based on the source?s instantaneous distance, diffuse-field or free-field amplitude attenuation is added in MATLAB, as is air absorption filtering. This virtual sound source method will be used for a variety of audio simulations and auditory experiments.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**12**	

Contents

List of Figures 43

Abstract 44

Acknowledgments 45

Student Bio 46

1. Introduction 47

2. Streaming Audio in MATLAB 47

3. Panning Algorithms 48

4. Signal Processing 49

5. Discussion 50

6. References 51

List of Figures

Figure 1. Plot of a symmetrical “dogbone” flight pattern...48

Abstract

Virtual sound source motion has been implemented in the U.S. Army Research Laboratory's Environment for Auditory Research, which contains a 57-channel spherical loudspeaker array located in a semi-anechoic chamber. Using the low-latency PortAudio application programming interface from the Psychophysics Toolbox Version 3, we are able to dynamically update 57 channels of streaming audio in real time using MATLAB for signal processing. Both Distance-Based Amplitude Panning (DBAP) and Vector Base Amplitude Panning (VBAP) have been implemented in MATLAB for controlling source motion. Sources are defined on a given path, such as a circle, ellipse, or the "dog bone" pattern often used in aviation. Although DBAP works convincingly for virtual sources located on the sphere defined by the loudspeaker array, VBAP is needed to position sources outside the array. Source motion paths are defined parametrically with respect to time, and the playback buffer updates the panned position every 11.5 ms. Based on the source's instantaneous distance, diffuse-field or free-field amplitude attenuation is added in MATLAB, as is air absorption filtering. This virtual sound source method will be used for a variety of audio simulations and auditory experiments.

Acknowledgments

Many thanks to my advisor, Mark Ericson, for giving me the tools and information needed to quickly get the ropes of an incredible audio reproduction facility at the Environment for Auditory Research. Mark's hospitality and guidance were crucial to the quick completion of this research. Thanks most of all to my wife Laura, who graciously put up with me being gone to undertake this work.

Student Bio

I studied Music Technology at Northwestern University, graduating as valedictorian of the School of Music in 2008. In 2009, I was awarded the Gates Cambridge Scholarship to fund a postgraduate degree at the University of Cambridge. I read for a Master of Philosophy (MPhil) in Physics at the Cavendish Laboratory, working in archeological acoustics. I am currently a second-year doctoral student at New York University, where I am working as part of the Music and Audio Research Laboratory's 3-D Audio group. My future plans include, but are not limited to, learning and helping others to learn.

1. Introduction

The Environment for Auditory Research is a state-of-the-art facility designed for a wide range of basic and applied auditory research (*1*). Its multiple loudspeaker configurations can create immersive audio simulations designed to test human auditory perception, detection thresholds, and angular and distance localization. One of its three main reproduction environments, the Sphere Room, contains a 57-channel spherical loudspeaker array, allowing a dense reproduction field along both azimuth and elevation dimensions.

The Sphere Room allows for easy reproduction of static soundscape simulations through a simple multichannel audio editor, such as Adobe Audition. However, the system must reproduce both static and moving sources to convincingly simulate real-world environments. Rendering a moving source via individual pans between 57 channels becomes quite tedious in a software environment designed for static sources. In Chowning's pioneering work on source motion, a signal-processing system was designed to precompute synthesized signals and change amplitudes on a quadraphonic reproduction system (*2*). This method has been adapted in the Sphere Room to precompute channel gains over time for a given source path and panning algorithm with recorded rather than synthesized source signals. The array of channel gains is then used to move the source by updating each loudspeaker's gain with real-time streaming audio. While this system was developed for experiments in the Sphere Room, it can also simulate source motion in less dense audio reproduction facilities.

2. Streaming Audio in MATLAB

MATLAB was used to implement the source motion system, which allowed low-level control of panning and signal processing. MATLAB interacts with the Sphere Room's RME Hammerfall DSP audio interface via the PortAudio application programming interface available through MATLAB's Psychophysics Toolbox Version 3 (PTB-3) extensions (*3*). PortAudio allows high-fidelity, low-latency audio control using static or streaming buffers. The streaming algorithm works by initiating an audio buffer and then continually appending small amounts of audio data onto the end of the buffer while playback is active.

The creators of PTB-3 recommend that each append to the buffer be no longer than half the latency of the audio system. Although lower latency allows the system to handle more iterations of the streaming loop in a given amount of time, the latency must allow time to process all real-time signal operations or the buffer will underflow. A series of tests using white noise and sinusoidal signals showed that the Hammerfall's internal buffer size should be set to 1024 samples, which corresponds to a system latency of 23 ms. This allows the streaming algorithm to update every 11.5 ms without underflowing. This update speed is sufficient to provide perceptually smooth panning paths across hundreds of individual positions. Multiple sources can be processed simultaneously, but the buffer size must be increased proportionally to handle real-time operations for each source.

3. Panning Algorithms

The Spatial Audio MATLAB Toolbox (*4*) was used for panning the virtual sources within the Sphere Room. This toolbox contains functions for both Distance-Based Amplitude Panning (DBAP, also called Vector Distance Panning in the toolbox) and Vector Base Amplitude Panning (VBAP). For both panning algorithms, the source's motion path was defined parametrically with respect to time. Test paths included a circle, ellipse, and the "dogbone" flight pattern often used in aviation—two long parallel lines with a larger curve at the end where the pilot turns around to come back along the other side, as shown in figure 1.

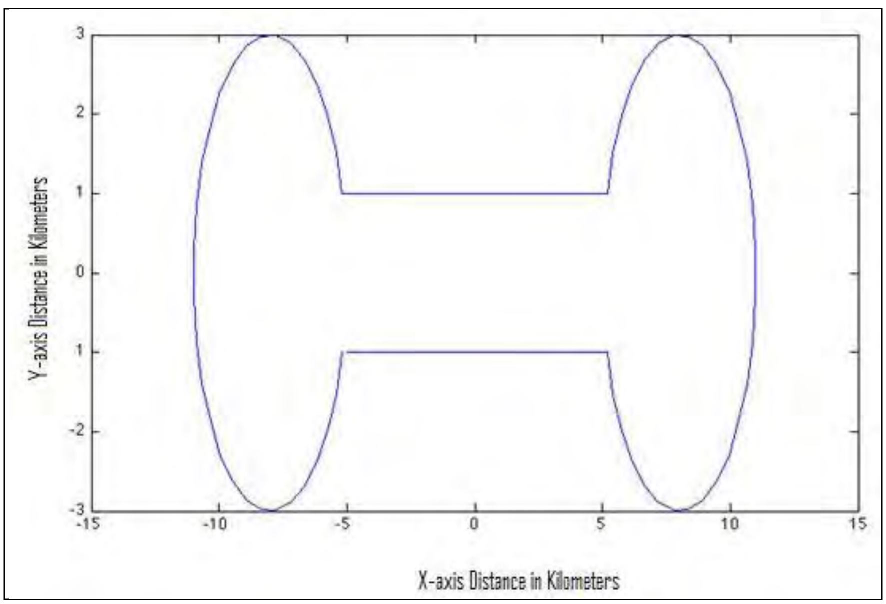

Figure 1. Plot of a symmetrical "dogbone" flight pattern.

DBAP pans a sound by simultaneously manipulating the gains of all of a system's loudspeakers based on only the source's distance to that loudspeaker and disregarding listener position (*5*). DBAP is relatively new but has performed well in subjective tests against more standard spatialization techniques, such as VBAP and Ambisonics (*6*). Because DBAP works for any number of loudspeakers at a single moment, it was the first panning algorithm implemented. DBAP panning gives a convincing impression of virtual sources along the sphere represented by the loudspeaker array. Source motion implemented with DBAP is smooth with a widened source impression because of the additional loudspeakers used to produce each virtual source. However, the drawback of DBAP is that it cannot accurately represent positions outside the loudspeaker array, since the difference between the loudspeakers' individual distances to a faraway source becomes negligible. While the creators of DBAP suggest other workarounds for such situations, such as projecting a distant source onto the loudspeaker array and then applying attenuation effects, it was decided to use only DBAP for virtual sources within the bounds of the loudspeaker array itself.

VBAP (*7*) is a robustly developed panning algorithm based on the following principle: given a set of three linearly independent vectors $\vec{a}$, $\vec{b}$, and $\vec{c}$ arranged in a triangle, any fourth vector $\vec{d}$, the desired source location, within that triangle may be created by a linear combination of the initial three vectors:

$$l_1\vec{a} + l_2\vec{b} + l_3\vec{c} = \vec{d}\ . \qquad (1)$$

If the first three vectors represent the positions of three loudspeakers arranged in a triangle, then a virtual source may be placed within that triangle by scaling and adding the first three vectors together. The coefficients of each vector thus become gain factors for each of the three loudspeakers in the triangle. While the Spatial Audio MATLAB Toolbox contained a function for calculating the VBAP gains for any given triangle of loudspeaker vectors, unconstrained source motion would require the system to automatically find which triangle the source vector intersected at any given point in time.

Sunday's algorithm (*8*) was implemented to calculate the vector-triangle intersections. The 57 loudspeakers in the Sphere Room define 110 distinct triangles, using the smallest angles possible in each case. These triangles' position vectors each define a unique plane, and the intersection algorithm first determines whether the source vector intersects that triangle's plane. If it intersects, the algorithm finds the plane's parametric coordinates **s** and **t** that determine the point of intersection. If the sum of **s** and **t** is between 0 and 1, then the vector intersects the given triangle. Finding these parameters requires only five distinct dot-product calculations, and the normal vectors of each plane may be precomputed and stored to optimize computation for static loudspeaker arrays. Because of this optimization, this algorithm is extremely efficient for VBAP systems whose loudspeaker triangles usually do not change position. The system then returns the current loudspeaker triangle based on the source's position and uses the Spatial Audio MATLAB Toolbox's VBAP function to calculate the appropriate gains within the loudspeaker triangle. These gains are then applied to the current point in the output signal and appended to the corresponding channels of the streaming 57-channel array via PortAudio.

4. Signal Processing

The method just described was initially intended for the reproduction of high-quality spatial recordings of aviation vehicles that were made available along with detailed x-y-z position data over time. When these data are used, reproduction may be perfectly adapted to a given recording, calculating loudspeaker gains in response to the recorded source's interpolated movement over time. This method maintains physical accuracy because distance-based attenuation, low-pass filtering due to air absorption, and Doppler effects are all contained within the original recording.

To make the system more flexible, however, we have begun to add additional signal processing capabilities to allow movement of an arbitrary signal. The current system has options of including free-field (inverse square) or diffuse-field (inverse alone) amplitude attenuation. This attenuation is added in real time to the streaming audio, although the attenuation coefficients are precomputed for each panning point once the source's motion path is defined. For air absorption filtering, following the suggestion of Huopaniemi (*9*), the yulewalk function from the MATLAB Signal Processing Toolbox (*10*) was implemented to reverse-engineer a one-pole filter based on known frequency-based attenuation data. Using the ANSI standard (*11*), we altered the system to find attenuation by octave bands as a function of distance and humidity, interpolating between known values if necessary. The filter coefficients are precomputed for the source path, but the filter can be applied to each 11.5 ms "chunk" of audio in real time without causing an underflow.

5. Discussion

The method outlined in the previous section incorporates streaming audio to allow low-latency virtual source motion in a large spherical loudspeaker array. DBAP and VBAP are both implemented, although VBAP is only needed for large-scale simulations. This method can be used with high-quality recordings to reproduce the exact path taken by an object recorded in motion. Amplitude attenuation and air absorption filtering have been added to allow flexible control of arbitrary source signals.

The drawback of this approach is that it makes sense for only signals whose audio content does not change based on their movement. While this might be the case for a person talking while walking at a low speed or even a white noise burst, which is fairly abstract in the first place, it does not really make sense for recordings of vehicles, whose engines' outputs vary greatly with velocity, not to mention the difficulty of modeling sound radiation patterns and source orientations. It is also difficult to obtain high-quality recordings of vehicles moving at a constant velocity, since the microphone is usually contained within the vehicle rather than outside it. An outside recording of a vehicle's engine idling may be used with this method to obtain variable results, but such a simulation will not sound convincing to an experienced listener. The inclusion of a flexible Doppler shift based on the source's relative velocity will be a future goal, although as mentioned previously, the sources that move fast enough to create a significant Doppler shift would usually be used in functions tailored specifically to the motion path of the recorded source. For slower-moving objects, the Doppler shift would have a negligible effect on the perception of the moving sound source.

6. References

1. Henry, P.; Amrein, B.; Ericson, M. The Environment for Auditory Research. *Acoustics Today* **2009**, *5* (3), 9–16.

2. Chowning, J. The Simulation of Moving Sources. *J. Audio Eng. Soc.* **1971**, *19*, 2–6.

3. Kleiner, M.; Brainard, D.; Pelli, D. What's New in Psychtoolbox-3? *Perception* [Online] **2007,** *36*, ECVP Abstract Supplement. http://www.perceptionweb.com/abstract.cgi?id=v070821 (accessed July 27, 2011).

4. Kostadinov, D.; Reiss, J. Spatial Audio MATLAB Toolbox, Centre for Digital Music, Queen Mary University of London, 2009. http://www.isophonics.net/content/spatial-audio-matlab-toolbox (accessed July 27, 2011).

5. Lossius, T.; Baltazar, P.; de la Hogue, T. DBAP - Distance-Based Amplitude Panning. *Proceedings of the 2009 International Computer Music Conference*, 2009.

6. Kostadinov, D.; Reiss, J.; Mladenov, V. Evaluation of Distance Based Amplitude Panning for Spatial Audio. *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2010.

7. Pulkki, V. Virtual Sound Source Positioning Using Vector Base Amplitude Panning. *J. Audio Eng. Soc*. **1997**, *45*, 456–466.

8. Sunday, D. Intersections of Rays, Segments, Planes and Triangles in 3D, 2003. http://www.softsurfer.com/Archive/algorithm_0105 (accessed July 27, 2011).

9. Huopaniemi, J.; Savioja, L.; Karjalainen, M. Modeling of Reflections and Air Absorption in Acoustical Spaces - A Digital Filter Design Approach. *IEEE 1997 Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, New York, 1997.

10. MATLAB, Signal Processing Toolbox, MathWorks Inc. http://www.mathworks.com/products/signal/ (accessed July 27, 2011).

11. American National Standard (ANS) 21.26-1995, Method for Calculation of the Absorption of Sound by the Atmosphere, 1995.